alpaka Parallel Programming – Online Tutorial
Lecture 00 – Getting Started with alpaka
Lesson 01: Introduction to alpaka
alpaka – Abstraction Library for Parallel Kernel Acceleration

Alpaka is...

• A parallel programming library: Accelerate your code by exploiting your hardware’s parallelism!
• An abstraction library: Create portable code that runs on CPUs and GPUs!
• Free & open-source software
Lesson 01: Introduction to alpaka

Programming with alpaka

- C++ only!
- Header-only library: No additional runtime dependency introduced
- Modern library: alpaka is written entirely in C++14
- Supports a wide range of modern C++ compilers (g++, clang++, Apple LLVM, MS Visual Studio)
- Portable across operating systems: Linux, macOS, Windows are supported
Lesson 01: Introduction to alpaka

alpaka’s purpose

Without alpaka

- Multiple hardware types commonly used (CPUs, GPUs, ...)
- Increasingly heterogeneous hardware configurations available
- Platforms not inter-operable → parallel programs not easily portable

alpaka: one API to rule them all

- Abstraction (not hiding!) of the underlying hardware & software platforms
- Code needs only minor adjustments to support different accelerators
Lesson 01: Introduction to alpaka

alpaka in the wild – example use case

PIConGPU: https://github.com/ComputationalRadiationPhysics/picongpu

- Fully relativistic, manycore, 3D3V particle-in-cell (PIC) code
- Central algorithm in plasma physics
- Scalable to more than 18,000 GPUs
- Developed at Helmholtz-Zentrum Dresden-Rossendorf
alpaka is free software (MPL 2.0). Find us on GitHub!

Our GitHub organization: https://www.github.com/alpaka-group
- Contains all alpaka-related projects, documentation, samples, ...
- New contributors welcome!

The library: https://www.github.com/alpaka-group/alpaka
- Full source code
- Issue tracker
- Installation instructions
- Small examples
Lesson 01: Introduction to alpaka

If you use alpaka for your research, please cite one of the following publications:


